A Systematic Procedure for Comparing Template-based Gesture Recognizers

Mehdi Ousmer[1[0000−0002−0222−0029]], Arthur Sluÿters[1[0000−0003−0804−0106]], Nathan Magrofuoco[1[0000−0003−4854−8502]], Paolo Roselli[2[0000−0003−3834−9358]], and Jean Vanderdonckt[1[0000−0003−3275−3333]]

[1] Université catholique de Louvain, LouRIM, Louvain-la-Neuve, Belgium
{mehdi.ousmer,arthur.sluyters,nathan.magrofuoco
jean.vanderdonckt}@uclouvain.be
[2] Università degli Studi di Roma, Roma, Italy
{roselli}@mat.uniroma2.it

Abstract. To consistently compare gesture recognizers under identical conditions, a systematic procedure for comparative testing should investigate how the number of templates, the number of sampling points, the number of fingers, and their configuration with other hand parameters such as hand joints, palm, and fingertips impact performance. This paper defines a systematic procedure for comparing recognizers using a series of test definitions, *i.e.* an ordered list of test cases with controlled variables common to all test cases. For each test case, its accuracy is measured by the recognition rate and its responsiveness by the execution time. This procedure is applied to six state-of-the-art template-based gesture recognizers on SHREC2019, a gesture dataset that contains simple and complex hand gestures tested and is widely used in the literature for competition in a user-independent scenario, and on Jackknife-lm, another challenging dataset. The results of the procedure identify the configurations in which each recognizer is the most accurate or the fastest.

Keywords: Gestural interaction · gesture recognizer · hand gesture recognition · real-time recognition · template matching.

1 Introduction

Miniaturized sensors are today incorporated into almost any everyday object or wearable object, such as smart watches and smart glasses, offering new sources of input for new forms of interaction [3]. Touchless user interfaces allow end users to view, control and manipulate any type of digital content [18], such as an object, an item, a scene, without physically touching the device, with fingers [7], hands [1, 34]. They are explored in a wide range of demanding contexts of use to the point where touchless interaction becomes a requirement. A comparative study is a research approach to evaluate a group of two or more instances of a tool, process, or software system. It relies on a framework to compare different systems that must be described in detail [17]. It is a test that differs from *benchmarking*, which evaluates performance against standard values on a group of metrics. While a significant body of knowledge exists for gesture recognition using sophisticated Machine Learning (ML) techniques, this article focuses on the family of template matching recognizers, which recognize a candidate gesture that matches a gesture class represented by a limited set of training templates. We consider this family of recognizers for their important properties [19]:

1. End users wish to interact by gesture in real time with their associated data with a discrete or continuous control. This interaction requires a low response time (*e.g.*, 0.1 s according to Nielsen [21]) and high accuracy (*e.g.*, a recognition rate of $\geq 90\%$ [20, 31]).
2. For designers, gesture vocabularies should be straightforward to design, to prototype [4], to edit, and to expand, particularly for user-defined gestures [32].
3. For a developer, these recognizers do not require extensive expertise in artificial intelligence as they are easy to train, modify, understand, compute, and interpret [6]. Their incorporation into a development cycle is clear. Pattern matching based on templates is widely used in system development [6].

The remainder of this paper is structured as follows: an introduction addresses the topic of existing surveys and other comparative testing focusing on the Leap Motion Controller (LMC). Section 3 presents some related works of comparative studies. Section 4 defines and applies the systematic procedure for comparative testing. Section 5 discusses the results. Finally, the document concludes with future research efforts and perspectives.

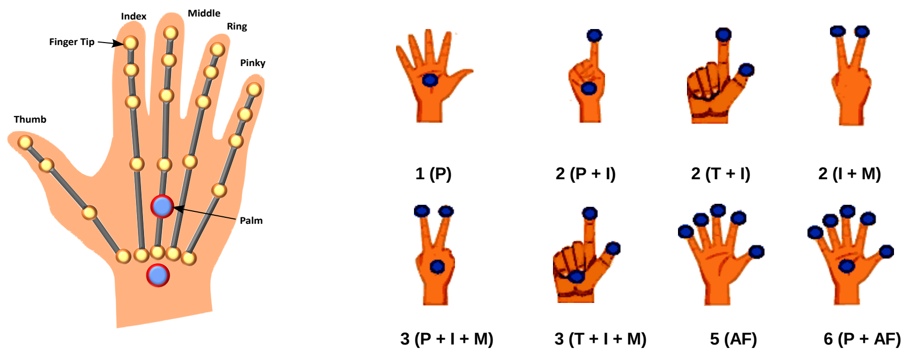
2 Related Work

Much work has been done on gesture recognition and particularly during the last decades, mainly on algorithms to effectively and efficiently recognize 2D gestures [5, 11, 28, 26] (see [18] for a survey), then 3D gestures in general [33] and for LMC in particular [14], mainly inspired by Artificial Intelligence (AI), Machine Learning (ML) [14] and computer vision [8, 23].

At the same time, many comparative studies were carried out in gesture recognition and related fields. The comparative study of Khan *et al.* [16] concerns various vision-based hand gesture recognition systems based on three main characteristics: segmentation, detection of features, and extraction phases. While another comparative study evaluates the effectiveness of a proposed feature selector method by comparing it with three other methods based on four performance measures and the prediction time; in order to demonstrate the impact of the feature selector on the performance of data fusion in activity recognition [30]. A comparative study between different devices to evaluate them in a game design context [15]. Despite the presence of numerous comparative testing for gesture recognition and related topics, there are none for hand gesture recognition using an LMC. Ferrer *et al.* [13] made a comparative study of a number of features for predicting human motion based on the minimum curvature variance. Although there are many comparative trials for gesture recognition and related topics, there are none for hand gesture recognition using the LMC.

3 Evaluation

The comparative testing performed in this paper aims at filling this gap in the literature. As such, we selected two available datasets for a number of reasons, including reproducibility. We evaluated the described recognizers in a user-independent scenario. We tested them on full hand gestures provided by the LMC skeletal hand model (Figure 1a) to show the efficiency of these recognizers.



(a) LMC Hand model[12] (b) The hand joints selected for the testing.

Fig. 1: Overview of the LMC Hand model and joints

3.1 Experiment

Design. Our evaluation was within factors with five independent variables:

1. Recognizer: nominal variable with 6 conditions, representing the various recognizers implemented for recognizing 3D gestures $\$P^3+$ [22], $\$F$[22], Jackknife[25] and $3Cent$[10]. And the two new recognizers which are described in Appendix A: $\$P^3 + X$ and *PennyPincher3D*.
2. Dataset: nominal variable with 2 condition, representing the datasets considered, *i.e.* SHREC2019 [9] and Jackknife-LM [25] described in Appendix B.
3. Joints: Nominal variable with 8 conditions, representing the hand joints used. Since the fingers contain several joints, we decided to use the information provided by the tips of the fingers (Figure 1b): **"1(P)"** = {Palm}, **"2(P+I)"** = {Palm, Index}, **"2(T+I)"**={Thumb, Index}, **"2(I+M)"** = {Index, Middle} , **"3(P+I+M)"** ={Palm, Index, Middle} , **"3(T+I+M)"** ={Thumb, Index, Middle}, **"5(AF)"** = {Thumb, Index, Middle, Ring, Pinky} , **"6(P+AF)"** = {Palm, Thumb, Index, Middle, Ring, Pinky}.
4. Number of Templates: numerical variable with 5 conditions, representing the number of templates per gesture for training: T={1, 2, 4, 8, 16}.
5. Sampling: numerical variable with 5 values representing the number of points per gesture: N={4, 8, 16, 32, 64}.

Apparatus. We used a hexa-core Intel Core i7 2.20 GHz CPU and a Windows 10 Home Edition operating system. The RAM was 16 GB DDR4 memory with 2400 MHz.

3.2 Procedure and Measures

We compute the *recognition rate* (computed as the ratio of positive recognitions divided by the total number of trials) for the 6 (Recognizer) × 2 (Dataset) = 12 basic configurations following the typical method used in the literature to evaluate gesture recognizers [4, 5, 32, 29, 28, 2, 22]: the *user-independent scenario* evaluates the recognition on gestures produced by users who are different from those used for training the recognizer. In this scenario, the basic configurations are refined depending on A, the number of joints, on T, the number of templates and depending on N, the number of resampling points to train the recognizer. A template is randomly selected for each gesture class from all participants and saved for testing. Then, a training set is obtained by randomly choosing T templates for each gesture class for all remaining users. They should be different from the templates previously selected for testing. Then, the recognizer is trained on the resulting training set. This operation is repeated R=100 times for each template of the T set.

4 Results and Discussion

Overall, we performed 2 (Dataset) × 8 (Joints) × 5 (Sampling) × 5 (Number of Templates) × 100 (repetitions) × 6 (Recognizer) = 240,000 recognition trials for each dataset. In the end, the number of recognized gestures is averaged to get the recognition rate and formatted as a percentage. We use GraphPad Prism to perform statistical computations.

4.1 SHREC2019 Dataset

Overall Recognition Rate. Figure 2 shows the average recognition rate for all tests under all conditions, with the recognizers displayed in descending order. The $\$P^3+$ has the best average recognition rate (M=85.90%, SD=15.44%), followed by $\$P^3 + X$ and Jackknife with, respectively (M=84.90%, SD=16.45%) and (M=82.11%, SD=14.06%), while the average value of the recognition rate

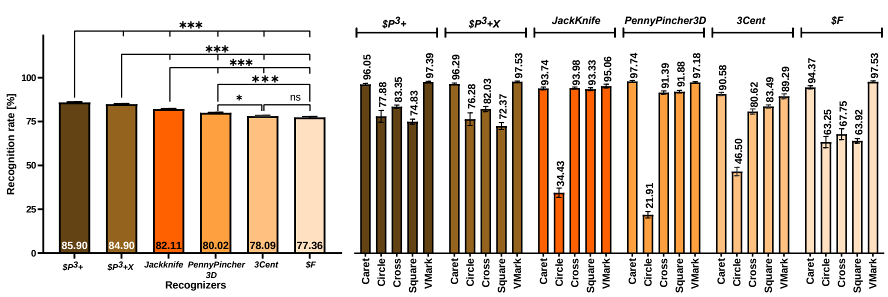


Fig. 2: Recognition rates of all recognizers for the SHREC2019 for all conditions. Error bars show a confidence interval of 95%.

of *PennyPincher3D* is equal to (M=80.02%, SD=12.38%), for the last two recognizers, the average rates do not exceed 80% with 3*Cent* (M=78.09%, SD = 18.12%) and \$$F$($M$=77.36%, SD=19.27%). We calculated the four normality tests for the Recognizer variable: K sample's Anderson-Darling test, D'Agostino's K2 test and Kolmogorov-Smirnov's KS. None of the recognition rates followed a normal distribution, while for the Shapiro-Wilk test, the number of samples was too large to test for normality. Next, we calculated a Kruskal-Wallis test with Dunn's multiple comparisons on the measures. The overall difference between the recognizers is very significant (p<.001***). Figure 2 shows that \$$P^3$+ is the most accurate Recognizer, and it significantly outclasses all other recognizers. \$$P^3$+ is better than \$$F$ with a highly significant difference (Z=48.84, p<.001***) and significantly better than Jackknife (Z=27.86, p<.001***). Also, \$$P^3$+ significantly outperforms \$$P^3+X$ (Z=5.247, p<.001***). However, \$$F$ is not significantly different from 3*Cent* (Z=1.928, $n.s.$), while this later is significantly outperformed by *PennnyPincher3D* (Z=3.055, p<.05*).

Furthermore, the overall recognition rate per gesture class indicates that the gestures "Caret" (/\) and "V-mark" (V) have average rates greater than or equal to 90%. Similarly to the average recognition rates of the other classes, they vary from one recognizer to another. While the "Cross" (X) is better recognized than the "Square" ([]) and the "Circle" (O) by the \$$F$, \$$P^3$+ and \$$P^3+X$ recognizers. The best average recognition rate for "Cross" (X) gesture class is achieved by Jackknife (M=93.98%); this value goes down respectively to (M=83.35%) for \$$P^3$+ and to ($M$=67.75%) for \$$F$ with a difference of 10.53% and 26.45%. For Jackknife and *PennyPincher3D*, the "Square" ([])) and "Cross" (X) classes have similar average recognition rates. On the contrary, the gesture class "Circle" (O) has a very low average recognition rate for Jackknife (M=34.43%), *PennyPincher3D* (M=21.91%) and 3*Cent* (M=46.50%). Detailed recognition rate tables of each condition are reproduced in Appendix C.2. A color-coding scheme reveals to what extent the recognition rate satisfies the rate expected by end users: $\tau \geq$90% [20, 31]. Some recognizers hold few or almost no green cells, such as *PennyPincher3D*, 3*Cent* and \$$F$, whereas \$$P^3$+ and \$$P^3 + X$ have frequent green cells, indicating that many conditions tested gave high recognition rates. Since the goal is to perform a comparative test of recognizers in the SHREC2019 dataset, we refined the results by relaxing the aforementioned constraint: $\tau \geq$80%.

Recognition Rates by Number of Joints. Figure 3 shows the average recognition rates for each value of condition Joints (A). Each recognition rate is the average of the recognition rates of all conditions (T) and (N). The values vary between M=75.5% for \$$F$ in $A = 2\ (I + M)$ and M=87.04% for \$$P^3$+ in $A = 3\ (P + I + T)$. In general, the recognition rates do not vary much from one value of (A) to another. We see that \$$P^3$+ has the highest average recognition rate for $A = 3\ : (P + I + T)$ (M=87.04%) and \$$F$ the lowest for $A = 2\ : (M{+}I)$ (M=75.58%). Although, there are some conditions $A = 2(I{+}M)$ and $A = 3\,(P + I + T)$, where the average rate of \$$P^3 + X$ exceeds that of \$$P^3$+ (*e.g.* $A = 2\,(I + M)$ and $A = 3\,(P + I + M)$). Similarly, for some values of (A), 3*Cent* has lower recognition rates than \$$F$ (*e.g.* $A = 1\,(P)$ and $A = 2\,(T + I)$).

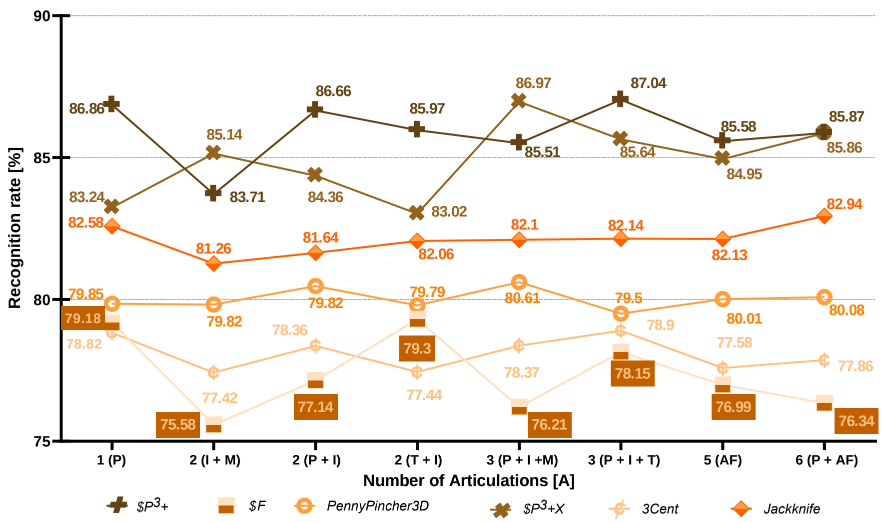


Fig. 3: Recognition rates of all recognizers for the SHREC2019 dataset [9], the plot shows average rates by joint $A = \{1\,(P), ..., 6\,(P + AF)\}$.

As a way to help understand the behavior of the recognizer with respect to both the number of points as well as the number of joints. Figure 11 in Appendix C.4 is an overview of recognizers ranked by recognition rate. This ranking is presented along two axes, the number of joints (A) and the number of sampling points (N). It shows which recognizers are better than the others for each pair of conditions (A, N). To respect the restriction on the recognition rate that we have defined above, all recognizers in this figure comply with two criteria: **1)** the recognizer must achieve a recognition rate $> 80\%$. **2)** The recognition rate must be the highest value among the conditions of (T). Globally, the position of the recognizers varies a lot according to the defined conditions, and few recognizers keep a constant ranking by varying the number of joints or the number of points (*e.g.*, *PennyPincher3D* for N=16 and N=32). For N=4, we observe that some recognizers do not appear, because of the first criterion defined, except for 3*Cent*, which has at least a recognition rate that respects the criterion. We notice that for several conditions, the $\$P^3+$ recognizer takes the first place, except for the condition where $A = 2\,(P + I)$ and $A = 2\,(I + M)$ where $\$P^3 + X$ is designated as the best recognizer. In contrast to $\$P^3+$, the *PennyPincher3D* often achieves the lowest recognition rate and is ranked last.

Recognition Rate for the Optimal Conditions. To evaluate the efficiency of the recognizers, we planned to test them in a precise scenario where the conditions are well-defined. For this reason, we determine which values of the variables Joints (A), Sampling (N), and Number of templates (T) allow an optimal comparison of the different recognizers in the recognition rate measure. We completed a de Borda ranking of all combinations of conditions for each recognizer, and the result is reported in Table 1. The de Borda ranking selects

Table 1: Global position of the recognizers for each condition and overall, according to de Borda method across all recognizers (the higher, the better).

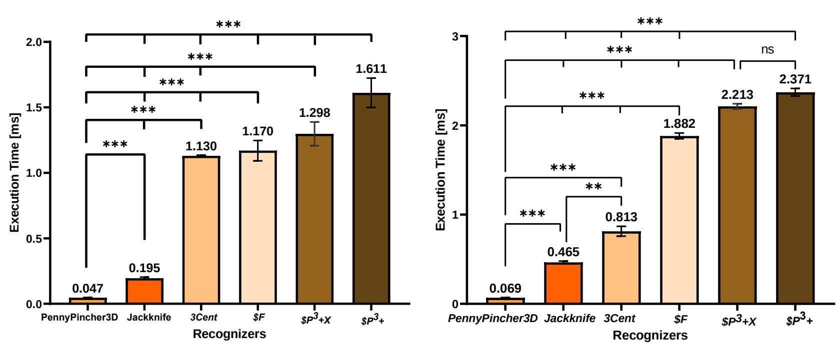
Condition				Recognizer (#Rank, Score)					
A	N	T	Overall	$\$P^3+$	$\$F$	*PP3D*	Jackknife	$\$P^3+X$	$3Cent$
2 (T + I)	32	16	#1, 1187	#1, 200	#1, 200	#2, 196	#1, 200	#1, 200	#2, 191
3 (P + I +M)	32	16	#2, 1180	#1, 200	#3, 184	#2, 196	#1, 200	#1, 200	#1, 200
2 (P + I)	32	16	#3, 1177	#1, 200	#2, 197	#2, 196	#2, 193	#1, 200	#2, 191
3 (P + I +M)	16	16	#4, 1175	#1, 200	#3, 184	#1, 200	#1, 200	#1, 200	#2, 191
2 (I + M)	32	16	#5, 1166	#2, 186	#1, 200	#2, 196	#2, 193	#1, 200	#2, 191
1 (P)	64	16	#6, 1161	#2, 186	#2, 197	#3, 185	#2, 193	#1, 200	#1, 200
2 (T + I)	16	16	#7, 1158	#1, 200	#1, 200	#3, 185	#1, 200	#2, 182	#2, 191
5 (AF)	32	16	#8, 1157	#1, 200	#2, 197	#2, 196	#2, 193	#1,200	#3, 171
6 (P + AF)	32	16	#9, 1153	#1, 200	#3, 184	#3, 185	#2, 193	#1, 200	#2, 191
3 (P + I + T)	32	16	#10, 1148	#1, 200	#2, 197	#3, 185	#2, 193	#2, 182	#2, 191
...	...	...	...	...	...	...	...	...	...

the best combination of conditions, the one with the highest overall Borda score. This implies that the elected combination of values has a high recognition rate with many recognizers. Furthermore, we notice that many conditions share the same rank and score in the ranking for each recognizer. This situation is due to a defined sensitivity value of 2% in the recognition rate. We determine the case of a tie between the last ranked combination of conditions and the following combination to be ranked on a specific recognizer, if the difference between their recognition rates is less than the defined sensitivity value. According to Table 1, the result of the Borda ranking gives this combination of conditions ($A = 2\ (T + I)/\ N$=32/ T=16) as the best.

Based on de Borda's results, we evaluate the recognizers on the SHREC2019 dataset under the elected conditions. The left part of Figure 10 in Appendix C.1 shows the average recognition rate for the defined conditions. According to these results, the $\$P^3+$ has the best average recognition rate (M=97.00%, SD=7.18%), followed by the $\$P^3 + X$ (M=95.20%, SD=8.59%) and the $\$F$ (M=94.20%, SD=9.55%), while *PennyPincher3D* is the least accurate recognizer (M=87.80%, SD=13.30%), for the other two recognizers, the average rate of Jackknife (M=93.2%, SD=10.34%) and $3Cent$ (M=90.80%, SD=13.16%). As for the overall recognition rate on all conditions in part 4.1, we calculated the four normality tests: K sample's Anderson-Darling test, Shapiro-Wilk W test, D'Agostino's K2 test and Kolmogorov-Smirnov's KS. None of the recognition rates followed a normal distribution. Then, we calculated a Kruskal-Wallis test with Dunn's multiple comparisons. The high difference between the recognizer $\$P^3+$ and *PennyPincher3D* (10%) is statistically highly significant (Z=5.736, p<.001***). Same thing for $\$P^3 + X$ that is significantly better than *PennyPincher3D* (Z=4.420, p<.001***). Moreover, $\$P^3+$ is significantly more accurate than $3Cent$ by 6.2% (Z=3.699, p<.01**). $\$F$ is superior by 6.4% to *PennyPincher3D* with a highly significant difference (Z=3.788, p<.01**). However, $\$P^3+$ is not significantly better than Jackknife (Z=2.581, *n.s.*), while this later is significantly better than *PennnyPincher3D* (Z=3.156, p<.05*).

In the same figure, the bar chart on the right shows the number of recognized gestures per gesture class on 100 repetitions, noting that all recognizers achieve a perfect score for at least one gesture class. Among the gesture classes in question, there is the "Caret" (/\) for *PennyPincher3D* and $3Cent$ recognizers, and also the "V-mark" (V) gesture class for $\$P^3+X$, $\$F$ recognizers. While $\$P^3+$ achieves an accuracy rate of ($\frac{100}{100}$) for two gesture classes : the"V-mark" (V) and the "Cross" (X). Although Jackknife achieves a flawless recognition for three gesture classes: 'Caret" (/\) , "Cross"(X) and 'V-mark" (V), its recognition of the "Circle"(O) gesture class is weak ($\frac{75}{100}$). This difficulty is also encountered by *PennyPincher3D* for the same gesture class ($\frac{59}{100}$) as for $3Cent$. However, the latter achieves the best accuracy rate for the "Square"([]) gesture class ($\frac{99}{100}$) which means that *PennyPincher3D* is well designed to recognize this gesture class, especially when the other recognizers perform less well in recognizing it.

Overall Conditions Execution Time. In Figure 4a, the average execution times of the different recognizers for all conditions varied between (M=0.047 ms, SD=0.060 ms) for the *PennyPincher3D* and (M=1.611 ms, SD=3.328 ms) for the $\$P^3+$. The execution times do not follow a normal distribution. We calculated a Kruskal-Wallis test which indicates a significant difference in the execution times of the recognizers. After that, the Dunn's multiple comparisons test shows significant differences for all pairs of recognizers except between $3Cent$ and $\$P^3+X$. The *PennyPincher3D* ($M$=0.047ms, SD=0.060ms) is significantly faster than other recognizers; It is significantly better than Jackknife (M=0.195 ms, SD=0.2457 ms) with a difference of 148μs (Z=73.62, p<.001***), and outperforms significantly the slowest recognizer $\$P^3+$ (Z=147.7, p<.001***).



(a) Execution times for all conditions. (b) Exec. times for the optimal conditions.

Fig. 4: Average execution times by recognizer. Error bars show a confidence interval of 95%.

Overall Execution Time by Number of Joints. The graphs in Figure 5 shows the execution times according to the values of the variable JOINTS (A). The smallest value in the graphic is the averaged execution time of *PennyPincher3D* for the condition $A = 1\,(P)$ (M=0.017 ms, SD=0.020 ms), while the longest execution time is achieved by $\$P^3$+ for $A = 6\,(P + AF)$ (M=3.204 ms, SD=5.936 ms). The *PennyPincher3D* is the fastest recognizer and is ahead of the Jackknife. According to the results, we distinguish two groups of recognizers with regard to their execution times. The one formed by Jackknife (M=0.195 ms, SD=0.246 ms) and *PennyPincher3D*, with a small slope, they are not much affected by the variation in the number of articulations. The second group consisting of $3Cent$(M=1.130 ms, SD=0.683 ms), $\$F$ (M=1.170 ms, SD=3.328 ms), $\$P^3 + X$($M$=1.298 ms, SD=2.731 ms) and $\$P^3$+ in order from fastest to slowest recognizer; All the recognizers are significantly impacted by the variation of Number of Joints (p<.001***).

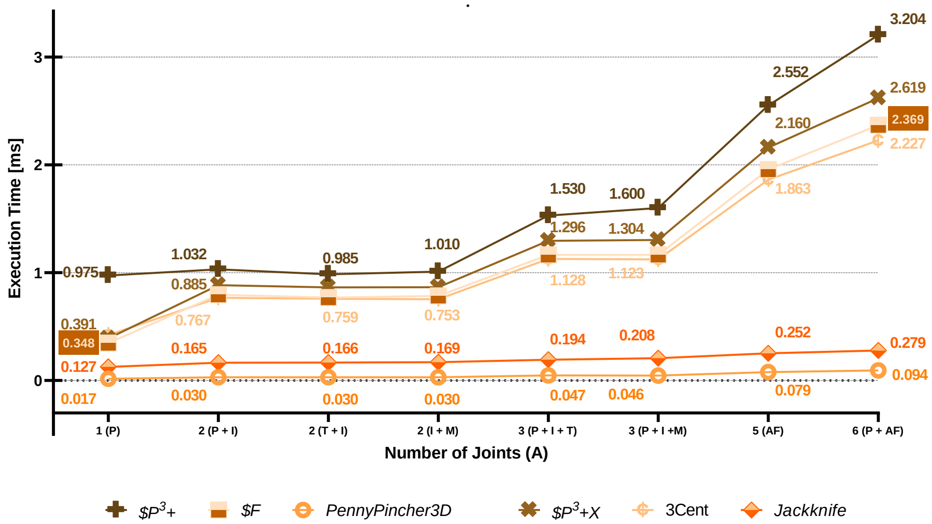
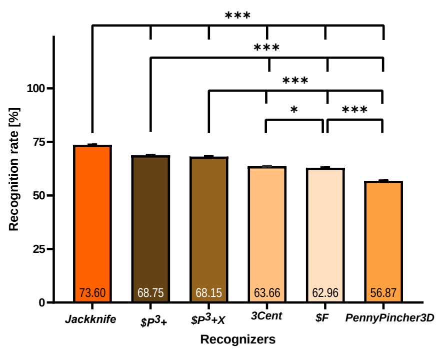


Fig. 5: Average execution times by recognizer per number of joints (A).

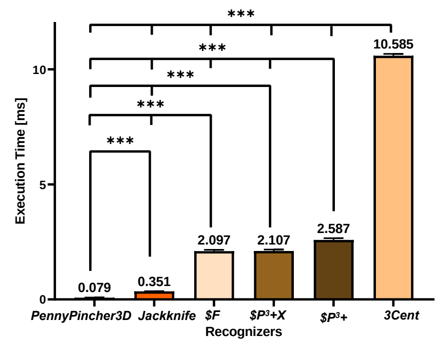
Execution Time for the Best Condition. Figure 4b shows the execution time of the recognizers for the best condition defined by the de Borda method. The recognizers appear in the same order as in the Figure 4a. *PennyPincher3D* remains the fastest (M=0.069 ms, SD=0.005 ms) and $\$P^3$+ the slowest ($M$=2.371 ms, SD=0.213 ms). The Kruskal-Wallis shows significant differences between all the recognizers except between the $\$P^3$+ and $\$P^3 + X$ (Z=1.910, $n.s.$).

4.2 Jackknife-LM Dataset

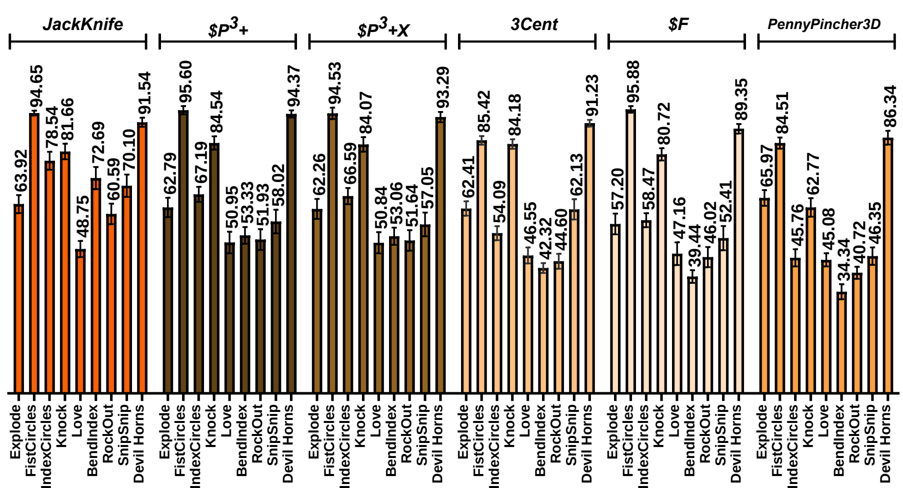
Overall Recognition Rate. The Jackknife-LM dataset has more gesture classes than SHREC2019, mainly complex gestures where different joints can move independently of the hand movement. Figure 6a sums up the averaged recognition results of each of the recognizers for all tests under all conditions. The results give an overview of the recognition with this dataset which are different from the results obtained for the SHREC2019. In general, none of the recognizers went beyond 80%. The Jackknife recognizer (M=73.60%, SD=19.75%) takes the lead and outperforms significantly other recognizers, followed by the $\$P^3+$ (M=68.75%, SD=19.73%), which is slightly better than $\$P^3+X$ (M=68.15%, SD=20.51%) by (0.60%). They are followed by $3Cent$ and $\$F$ with respectfully (M=63.66%, SD=18.29%) and (M=62.96%, SD=20.12%) which are significantly better than *PennyPincher3D* the least efficient of the recognizers (M=56.87%, SD=19.99%).



(a) Recognition rates for all conditions.



(b) Execution times for all conditions.



(c) Overall recognition rates for all conditions by gesture class.

Fig. 6: Recognition rates and execution times of all recognizers for the Jackknife-LM for user independent scenario. Error bars show a confidence interval of 95%.

Three gesture classes ("FistCircles", "Knock" and "Sideways") are well recognized for all the recognizers (*i.e.* $\tau \geq 80\%$ - Figure 6c, bottom). An exception to this is *PennyPincher3D* where the "Knock" gesture has an average recognition rate of (M=62, 77%). However, the "Love", "BendIndex", "DevilHorns", and "SnipSnip" have the worst average recognition rates for most recognizers (*i.e.* $\tau \leq 60\%$), except for Jackknife, where the "SnipSnip" and "BendIndex" are above 70%. These results indicate that for many recognizers, many conditions perform very well with gestures where the fingers remain static, regardless of hand movements. Whereas the recognition rate drops for many conditions with gestures that include finger movements. From the table in the Appendix C.3 which details the average recognition rates for the different conditions, the orange cells are the predominant ones in the table for the majority of the recognizers, which express a low recognition rate for many conditions $\tau \leq 80\%$ (*e.g. PennyPincher3D*, $3Cent$ and $\$F$). For *PennyPincher3D*, $3Cent$, no single condition achieves a 90% recognition rate. While for other recognizers, many conditions reach rates above 90% with $A = 5\,(AF)$ and $A = 6\,(P + AF)$, which denotes the ability to handle complex gestures under certain conditions.

Recognition Rates by Number of Joints. With respect to the average recognition rates by articulation in Figure 7, the lowest recognition rate is M=39.78% for $\$F$ in $A = 1\,(P)$, whereas the top recognition rate is M=87.78% for $\$P^3 + X$ in $A = 6\,(P + AF)$. The Jackknife is above other recognizers for the conditions with a reduced number of articulations ($A \leq 3$), then it is joined by $\$P^3+$ and $\$P^3 + X$ at $A = 5\,(AF)$. The average recognition rate increases as the number of joints increases for all of the recognizers, except for some particular cases, as for $3Cent$, *PennyPincher3D* and $\$P^3+$ whose rate drops between $A = 5\,(AF)$ and $A = 6\,(P + AF)$ (*e.g.* a drop of 2.6% for $3Cent$). However, $\$P^3 + X$, $\$P^3+$, $\$F$ recognition rates under conditions with the same number of joints vary depending on how the joints are configured (*e.g.*, A=2($P + I$), A=2($T + I$), A=2($I + M$)). Therefore, some joints are more relevant than others since they carry more information about gestures.

Overall Conditions Execution Time. Figure 6b shows the values of the average execution times for all conditions, which indicates the execution time for the various recognizers. There is a big difference between the execution time of *PennyPincher3D* (M=0.079 ms,SD=0.108 ms) and $3Cent$ (M=10.585 ms,SD=5.911 ms). The execution time includes the pre-processing time of the gesture and the recognition time. For $3Cent$, the excessive execution time is caused by the resampling function based on the Cubic Spline interpolation method. The execution times of the other recognizers are close to one another, $\$P^3+$ (M=2.587 ms, SD=4.371 ms), $\$F$ (M=2.097 ms, SD=4.184 ms) and $\$P^3 + X$ (M=2.107 ms, SD=4.371 ms) and there is no significant difference between the latter two (Z=1.772, $n.s.$). Jackknife, which has the highest recognition rate, is also a fast recognizer (M=0.351 ms,SD=0.508 ms). Thus, it makes it well suited to this dataset.

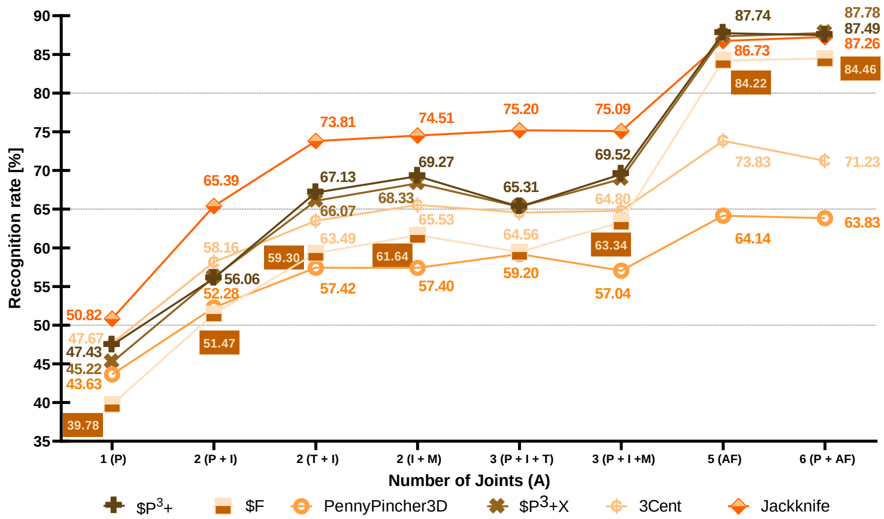


Fig. 7: Recognition rates of all recognizers for the Jackknife-LM dataset [25], the plot shows average rates by joint $A = \{1\,(P), ..., 6\,(P + AF)\}$.

5 Conclusion

The objective of this comparative study on LMC-based gestures is to fill a gap in the literature. For this reason, we evaluated in a user-independent scenario six gesture recognizers on two LMC-Based datasets composed of simple and complex gestures (*i.e.* SHREC2019 and Jackknife-LM). Overall, for the SHREC2019, $\$P^3+$ achieved the best recognition rate and is significantly better than the other recognizers and it was confirmed under particular optimal conditions, while $\$F$ is the worst for this dataset. We also noticed that some recognizers are more adapted to recognize certain gestures than others. For the Jackknife-LM dataset, the Jackknife recognizer achieves overall good recognition rates under certain conditions but does not satisfy the high accuracy property for many other conditions. We have seen that many recognizers are affected by the nature of the gesture performed. Furthermore, some recognizers are slow to process long gestures, making them less attractive. This procedure is a good contribution to designers who wish to choose a reliable and efficient recognizer, and thus guide them to meet their needs. The limitation of this study is that it only includes one type of gesture recognition algorithm based on the template matching technique. A solution to be implemented in the future would be to extend the tests with new machine learning algorithms and new gestures related to specific application fields by including more datasets.

Acknowledgements

Arthur Sluÿters is funded by the “Fonds de la Recherche Scientifique - FNRS” under Grant no. 40001931.

A The New Recognizers Considered in the Experiment

A.1 $P^3+X Recognizer

A variant of $P^3+ [27] that takes into account the direction-invariance by tracking conflicting templates (*i.e.*, templates of the same gesture but performed in different directions). If a gesture matches with a conflicting template, its direction is compared with the direction of each conflicting template, and the nearest one is chosen.

A.2 PennyPincher3D Recognizer

PennyPincher3D is an adaptation of the 2D recognizer *PennyPincher* [24]. The gestures are represented as a set of $N-1$ vectors linking between N equidistant points. The recognizer matches the candidate gesture with the template that maximizes a dissimilarity score, computed as the sum of the angles between the vectors. The computation relies on basic mathematical operations such as additions and multiplications. The gestures require just a resampling as prepossessing. This recognizer is scale- and position- invariant as most of the $-recognizers.

B The Datasets Considered in the Experiment

B.1 SHREC2019 [9]

The **SHREC2019** dataset [9] contains a sequence of 3D points and quaternions for each hand's joint designating one of five gesture classes (Figure 8): "Cross" (X), "Circle" (O), "V-mark" (V), "Caret" (/\), and "Square" ([])). It served in (SHREC) track, a contest on online gesture recognition to detect command gestures from hands' movements in a virtual reality context. The proposed dataset consists of 195 3D movements performed by 13 participants with the whole hand. The dataset contains unsegmented gestures. The training set and the testing set were merged to create a unique dataset in which, unnecessary hand movements were removed from the gestures.

Fig. 8: The SHREC2019 gesture classes.

B.2 Jackknife-LM [25]

The **Jackknife-LM (Jackknife-LeapMotion)** dataset [25] contains 3D complex gestures of the hand and saved as 3D skeleton which is provided by the LeapMotion. We used the segmented gestures composed by 360 samples of 9 different gesture classes for example "Fist Circles", "Snip Snip", "Explode"(Figure 9). It was used to test a rejection approach of non-gesture sequences from a continuous data stream. While segmented gestures make up the training set, authors employ unsegmented sessions of samples in the test[25].

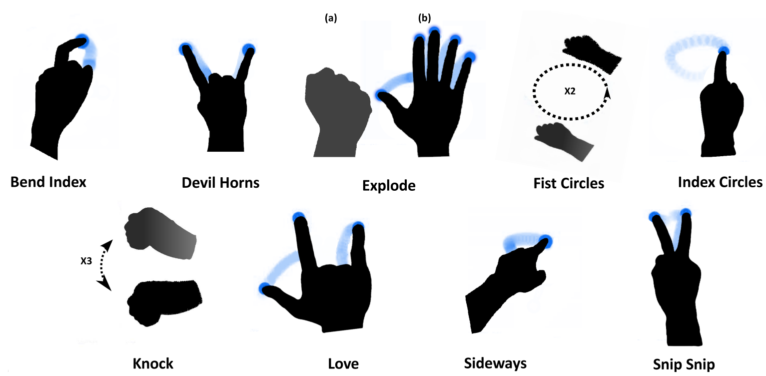


Fig. 9: The Jackknife-LM gesture classes [25].

C Recognition Rates

C.1 Recognition Rate for best condition for the SHREC2019

.

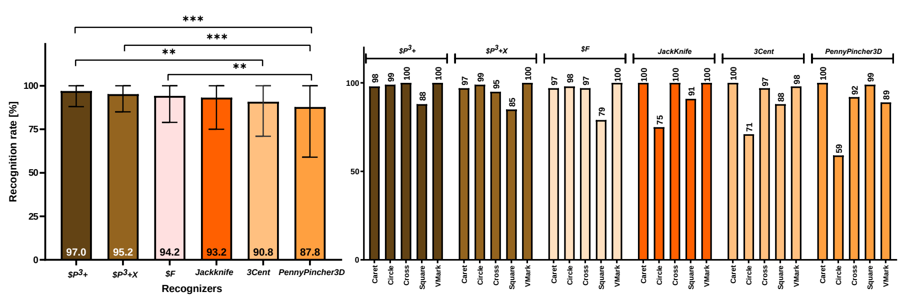


Fig. 10: Recognition rate (left) and the number of recognized gestures per class over 100 trials (right) of all recognizers for user independent scenario, for the optimal conditions defined by the de Borda ranking: A=2 $(T+I)$, N=32, T=16. Error bars show a confidence interval of 95%.

C.2 Recognition Rates Tables for the SHREC2019

Number of templates (T); Number of points (N)

$P³+

Configuration	N	1	2	4	8	16
1 (P)	4	71.8	73.8	77.8	81.4	82.8
1 (P)	8	79.8	78.4	84.6	88.2	91.4
1 (P)	16	84.0	92.4	90.2	96.4	94.6
1 (P)	32	85.4	90.4	91.4	94.2	93.8
1 (P)	64	78.8	88.0	93.0	94.0	94.8
2 (P + I)	4	71.4	75.6	79.6	79.0	83.0
2 (P + I)	8	77.4	82.4	87.2	87.2	89.0
2 (P + I)	16	82.4	91.0	94.4	93.8	94.2
2 (P + I)	32	83.6	89.0	92.0	95.0	96.2
2 (P + I)	64	82.8	85.2	89.4	91.6	94.2
2 (T + I)	4	70.4	74.8	74.4	80.2	86.2
2 (T + I)	8	75.4	80.2	84.2	84.6	90.2
2 (T + I)	16	78.8	89.0	91.4	91.6	96.2
2 (T + I)	32	82.4	90.0	92.0	96.2	97.0
2 (T + I)	64	79.0	85.6	90.0	92.8	96.6
2 (I + M)	4	72.4	71.0	74.6	76.8	76.6
2 (I + M)	8	75.0	79.6	84.2	84.4	88.2
2 (I + M)	16	79.8	84.2	91.4	89.6	93.8
2 (I + M)	32	78.6	84.0	90.8	94.8	94.6
2 (I + M)	64	75.0	82.8	85.6	90.4	94.6
3 (P + I + T)	4	68.8	76.0	78.4	79.6	81.4
3 (P + I + T)	8	79.4	79.8	84.4	88.4	92.2
3 (P + I + T)	16	85.6	91.8	91.8	93.6	95.4
3 (P + I + T)	32	85.2	91.2	93.0	94.0	96.8
3 (P + I + T)	64	83.6	86.2	90.8	94.6	94.0
3 (P + I +M)	4	70.0	74.6	77.6	77.4	73.8
3 (P + I +M)	8	77.4	79.6	86.0	86.2	92.2
3 (P + I +M)	16	81.0	88.4	92.6	94.0	96.0
3 (P + I +M)	32	81.2	86.6	91.0	95.0	95.6
3 (P + I +M)	64	80.6	85.2	90.0	92.2	93.6
5 (AF)	4	72.4	73.6	74.6	76.4	78.4
5 (AF)	8	77.6	81.0	83.6	87.2	93.4
5 (AF)	16	81.2	88.8	91.2	93.8	94.4
5 (AF)	32	83.8	90.2	92.6	94.2	96.2
5 (AF)	64	78.4	83.4	89.6	90.0	93.4
6 (P + AF)	4	69.2	73.2	74.2	77.8	75.6
6 (P + AF)	8	76.8	82.8	84.2	89.8	93.6
6 (P + AF)	16	81.8	90.4	92.8	93.4	96.8
6 (P + AF)	32	82.4	89.6	88.8	95.2	96.6
6 (P + AF)	64	78.6	86.0	90.0	93.8	93.4

$P³+X

Configuration	N	1	2	4	8	16
1 (P)	4	66.8	66.4	58.6	52.2	48.8
1 (P)	8	78.4	83.4	83.6	88.6	89.6
1 (P)	16	87.4	88.4	95.0	95.4	93.4
1 (P)	32	85.0	89.4	91.4	93.8	94.0
1 (P)	64	81.6	88.2	91.6	93.6	96.4
2 (P + I)	4	70.6	74.4	73.0	74.0	65.6
2 (P + I)	8	77.2	81.8	87.2	86.2	88.6
2 (P + I)	16	85.2	86.4	93.6	93.6	94.8
2 (P + I)	32	83.8	87.2	92.4	95.4	97.0
2 (P + I)	64	79.4	84.6	87.8	93.4	95.4
2 (T + I)	4	70.2	72.8	73.6	67.2	65.6
2 (T + I)	8	75.2	81.0	82.6	85.0	88.8
2 (T + I)	16	80.8	87.8	91.6	93.2	95.2
2 (T + I)	32	81.0	85.4	94.8	95.4	95.2
2 (T + I)	64	79.8	86.6	89.4	94.0	96.8
2 (I + M)	4	67.8	71.2	72.8	69.8	66.2
2 (I + M)	8	77.6	78.6	82.6	84.6	85.4
2 (I + M)	16	79.6	86.8	88.8	91.6	94.0
2 (I + M)	32	80.2	85.4	90.6	91.6	96.8
2 (I + M)	64	77.0	82.6	88.0	90.8	95.0
3 (P + I + T)	4	69.8	75.6	76.6	79.8	83.0
3 (P + I + T)	8	76.6	83.8	83.6	86.6	91.6
3 (P + I + T)	16	85.2	91.6	95.2	94.8	95.8
3 (P + I + T)	32	85.0	88.4	91.6	96.6	96.8
3 (P + I + T)	64	78.0	87.4	90.0	95.0	95.8
3 (P + I +M)	4	69.0	74.4	76.0	77.0	74.6
3 (P + I +M)	8	81.0	84.6	85.2	87.8	90.2
3 (P + I +M)	16	80.8	87.0	93.6	92.8	93.6
3 (P + I +M)	32	82.2	89.2	91.4	93.2	94.8
3 (P + I +M)	64	81.6	84.8	90.4	93.4	92.4
5 (AF)	4	70.0	72.6	74.2	76.8	73.0
5 (AF)	8	77.6	78.8	86.0	88.8	90.8
5 (AF)	16	83.0	88.6	90.2	93.6	94.0
5 (AF)	32	81.4	88.4	89.6	93.8	95.6
5 (AF)	64	77.8	85.8	87.0	92.0	94.4
6 (P + AF)	4	69.8	75.0	68.6	75.8	79.8
6 (P + AF)	8	76.4	81.4	88.8	89.6	95.4
6 (P + AF)	16	84.0	84.2	94.0	93.8	93.8
6 (P + AF)	32	82.6	90.2	91.0	92.6	96.4
6 (P + AF)	64	81.6	85.8	89.0	93.2	93.6

Jackknife

Configuration	N	1	2	4	8	16
1 (P)	4	74.0	78.6	82.0	83.8	80.4
1 (P)	8	71.6	76.2	79.8	86.2	87.8
1 (P)	16	74.6	82.8	85.2	89.4	92.8
1 (P)	32	74.8	79.4	87.6	89.4	90.8
1 (P)	64	74.8	81.0	84.4	86.0	91.0
2 (P + I)	4	73.2	76.6	82.4	79.2	81.2
2 (P + I)	8	71.0	76.0	76.2	82.2	87.4
2 (P + I)	16	72.8	81.4	85.2	91.8	90.8
2 (P + I)	32	75.6	77.8	84.0	90.0	92.4
2 (P + I)	64	73.4	80.0	83.6	87.4	89.4
2 (T + I)	4	74.8	79.4	81.6	83.6	79.4
2 (T + I)	8	66.2	75.8	77.4	81.6	85.2
2 (T + I)	16	75.6	80.8	86.2	88.8	93.6
2 (T + I)	32	73.0	80.8	85.8	88.6	93.2
2 (T + I)	64	74.8	80.6	82.8	87.2	94.8
2 (I + M)	4	72.4	78.0	80.8	81.8	82.8
2 (I + M)	8	65.2	77.6	75.8	84.4	86.0
2 (I + M)	16	73.4	78.8	86.4	88.6	93.0
2 (I + M)	32	75.2	79.0	82.6	90.8	92.6
2 (I + M)	64	74.2	78.8	81.8	83.8	87.8
3 (P + I + T)	4	75.2	76.6	81.0	79.4	81.2
3 (P + I + T)	8	69.4	76.2	80.2	80.4	85.2
3 (P + I + T)	16	75.4	82.8	86.2	91.8	93.8
3 (P + I + T)	32	74.8	80.6	86.0	87.6	91.4
3 (P + I + T)	64	75.4	81.0	84.6	87.2	90.0
3 (P + I +M)	4	74.8	80.2	81.6	83.8	84.0
3 (P + I +M)	8	68.4	74.4	81.6	83.0	88.6
3 (P + I +M)	16	73.4	78.6	85.0	90.0	94.4
3 (P + I +M)	32	73.4	82.6	87.2	89.8	93.6
3 (P + I +M)	64	73.0	76.0	81.8	86.6	86.8
5 (AF)	4	74.8	79.6	80.4	81.4	78.8
5 (AF)	8	67.2	76.0	81.0	85.2	86.6
5 (AF)	16	73.4	81.2	89.2	91.6	92.8
5 (AF)	32	78.6	82.6	85.0	88.2	92.8
5 (AF)	64	75.2	78.0	80.6	84.0	89.0
6 (P + AF)	4	76.8	79.8	79.8	83.4	83.4
6 (P + AF)	8	72.0	74.8	80.2	84.8	90.0
6 (P + AF)	16	74.4	80.0	85.8	90.6	91.6
6 (P + AF)	32	78.6	80.6	87.4	89.0	92.8
6 (P + AF)	64	78.0	79.8	83.8	88.0	88.0

3Cent

Configuration	N	1	2	4	8	16
1 (P)	4	64.0	73.4	78.4	79.8	84.0
1 (P)	8	68.6	74.8	82.2	88.0	90.6
1 (P)	16	64.0	72.6	80.2	84.2	92.8
1 (P)	32	64.4	73.4	80.8	83.8	92.4
1 (P)	64	66.4	72.0	79.8	87.4	92.6
2 (P + I)	4	63.6	68.8	74.2	79.2	84.6
2 (P + I)	8	66.0	73.0	81.4	84.8	89.2
2 (P + I)	16	68.8	73.8	80.2	85.2	91.4
2 (P + I)	32	67.8	73.8	80.2	84.0	90.0
2 (P + I)	64	66.0	74.8	80.2	86.2	91.8
2 (T + I)	4	65.4	69.0	76.2	79.0	82.4
2 (T + I)	8	65.6	73.2	77.8	86.0	89.6
2 (T + I)	16	68.6	72.8	75.2	85.2	89.0
2 (T + I)	32	69.2	69.4	78.0	85.6	90.8
2 (T + I)	64	63.6	71.8	77.6	86.6	88.4
2 (I + M)	4	65.0	68.0	73.8	80.0	84.2
2 (I + M)	8	65.4	75.6	79.6	83.0	90.0
2 (I + M)	16	64.6	71.4	80.2	84.2	89.0
2 (I + M)	32	67.2	71.6	78.2	84.2	89.6
2 (I + M)	64	64.4	72.0	78.6	86.4	89.2
3 (P + I + T)	4	64.4	75.6	76.2	79.0	84.4
3 (P + I + T)	8	65.0	74.4	82.0	86.2	90.2
3 (P + I + T)	16	66.2	75.0	82.0	87.0	89.6
3 (P + I + T)	32	66.2	71.8	83.0	84.4	90.0
3 (P + I + T)	64	70.2	73.2	81.2	84.2	91.0
3 (P + I +M)	4	62.4	70.6	74.8	79.8	84.2
3 (P + I +M)	8	69.4	75.2	80.8	82.0	91.4
3 (P + I +M)	16	64.0	73.2	79.6	86.2	90.6
3 (P + I +M)	32	68.2	75.8	81.8	85.0	91.6
3 (P + I +M)	64	65.0	71.0	81.2	86.0	89.4
5 (AF)	4	63.0	69.4	77.6	81.2	85.4
5 (AF)	8	68.2	73.2	77.0	84.6	89.2
5 (AF)	16	65.6	70.2	81.0	87.2	90.2
5 (AF)	32	65.6	72.0	77.0	83.8	88.0
5 (AF)	64	65.6	71.6	78.4	84.6	89.8
6 (P + AF)	4	64.8	71.6	76.2	82.6	84.4
6 (P + AF)	8	66.4	73.0	82.8	85.8	89.8
6 (P + AF)	16	64.6	69.2	77.8	87.0	91.2
6 (P + AF)	32	66.0	74.8	75.0	83.4	89.6
6 (P + AF)	64	66.4	73.6	78.6	83.8	88.0

PennyPincher3D

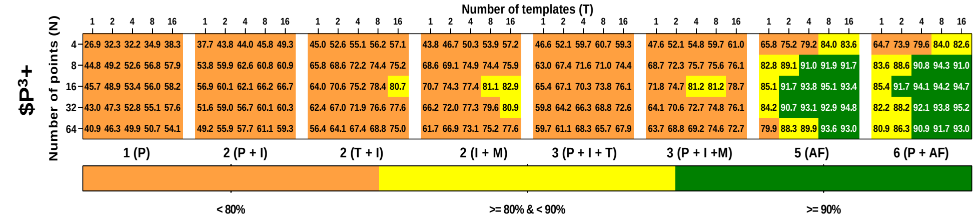
Configuration	N	1	2	4	8	16
1 (P)	4	67.0	66.6	74.2	75.8	77.2
1 (P)	8	76.4	78.2	80.6	83.4	84.8
1 (P)	16	77.4	79.4	82.8	84.4	85.0
1 (P)	32	76.6	79.2	85.8	84.8	86.8
1 (P)	64	76.4	80.6	82.6	84.8	85.4
2 (P + I)	4	67.0	74.0	75.8	74.0	78.4
2 (P + I)	8	76.2	79.0	81.4	82.2	85.2
2 (P + I)	16	77.4	81.2	81.4	84.0	87.2
2 (P + I)	32	78.6	79.8	82.0	84.6	88.0
2 (P + I)	64	76.8	77.8	82.8	86.4	90.6
2 (T + I)	4	67.4	72.4	74.2	78.8	76.2
2 (T + I)	8	77.0	79.6	82.4	80.6	84.6
2 (T + I)	16	77.0	78.0	79.4	82.0	85.6
2 (T + I)	32	77.6	80.0	81.4	85.4	87.8
2 (T + I)	64	76.6	79.0	81.0	83.2	87.6
2 (I + M)	4	69.2	73.4	73.2	77.2	77.4
2 (I + M)	8	78.2	79.4	81.4	82.8	84.8
2 (I + M)	16	75.6	79.4	79.0	82.2	87.4
2 (I + M)	32	75.4	77.6	82.0	85.6	86.4
2 (I + M)	64	77.0	79.0	82.6	83.8	85.6
3 (P + I + T)	4	70.8	76.2	74.8	76.6	75.0
3 (P + I + T)	8	77.6	78.6	81.4	79.8	82.0
3 (P + I + T)	16	76.4	80.4	79.4	82.8	83.2
3 (P + I + T)	32	76.8	78.6	80.2	82.8	85.4
3 (P + I + T)	64	76.8	78.6	81.8	85.8	85.8
3 (P + I +M)	4	72.2	76.8	77.4	77.2	77.0
3 (P + I +M)	8	79.0	79.0	81.2	82.8	84.8
3 (P + I +M)	16	77.4	80.0	81.0	83.2	89.0
3 (P + I +M)	32	77.8	79.4	79.8	83.0	86.6
3 (P + I +M)	64	77.6	79.0	81.0	83.8	89.2
5 (AF)	4	70.6	73.2	77.4	79.8	80.4
5 (AF)	8	75.2	78.6	81.2	81.6	84.2
5 (AF)	16	75.8	78.6	80.0	79.4	84.0
5 (AF)	32	76.4	80.0	81.6	82.0	87.2
5 (AF)	64	76.6	80.8	82.4	85.4	87.8
6 (P + AF)	4	72.2	76.6	75.4	78.6	78.4
6 (P + AF)	8	78.6	79.6	80.4	83.0	84.2
6 (P + AF)	16	77.2	79.2	79.2	83.2	81.8
6 (P + AF)	32	77.2	76.6	81.6	82.8	84.6
6 (P + AF)	64	77.4	79.2	80.6	85.0	89.4

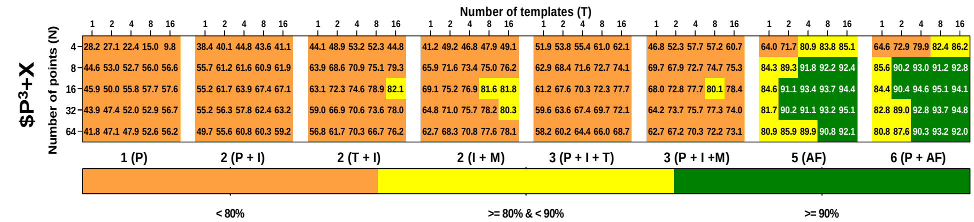
$F

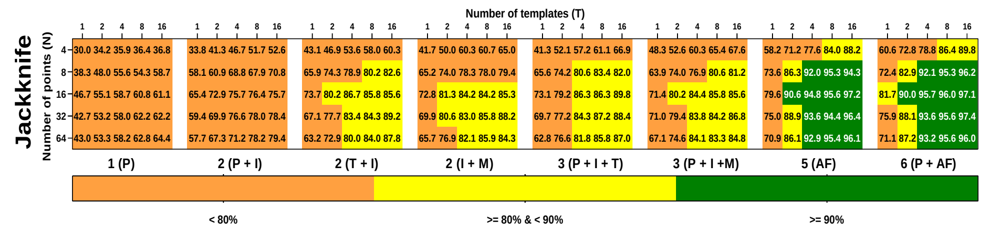
Configuration	N	1	2	4	8	16
1 (P)	4	58.6	59.0	62.0	65.6	66.6
1 (P)	8	64.8	76.4	80.6	80.6	87.0
1 (P)	16	77.6	82.4	82.6	87.0	87.2
1 (P)	32	77.6	84.2	87.2	90.0	91.6
1 (P)	64	78.6	81.6	88.0	91.8	90.8
2 (P + I)	4	53.6	56.2	60.2	62.8	59.6
2 (P + I)	8	70.4	72.2	74.6	79.0	82.6
2 (P + I)	16	75.0	80.4	81.4	84.0	89.0
2 (P + I)	32	77.8	83.0	86.6	88.6	91.2
2 (P + I)	64	79.4	81.4	83.4	88.2	88.0
2 (T + I)	4	50.8	61.8	65.4	62.2	68.2
2 (T + I)	8	68.6	78.2	75.0	83.6	84.6
2 (T + I)	16	77.8	80.6	87.4	90.8	93.4
2 (T + I)	32	77.8	82.6	86.4	89.8	94.2
2 (T + I)	64	76.8	79.6	84.8	90.2	91.8
2 (I + M)	4	52.6	54.0	55.2	52.8	58.8
2 (I + M)	8	64.8	70.8	74.2	81.8	83.2
2 (I + M)	16	71.4	75.2	82.4	89.0	90.0
2 (I + M)	32	74.0	77.6	81.6	91.6	92.8
2 (I + M)	64	71.6	78.6	84.2	89.6	91.8
3 (P + I + T)	4	55.0	60.8	59.8	61.8	66.0
3 (P + I + T)	8	70.4	74.2	78.0	79.4	80.2
3 (P + I + T)	16	74.6	84.4	88.2	86.2	88.4
3 (P + I + T)	32	77.2	85.4	85.8	88.6	90.6
3 (P + I + T)	64	77.4	82.0	86.0	87.0	86.4
3 (P + I +M)	4	53.2	57.8	60.4	57.6	56.6
3 (P + I +M)	8	67.0	71.0	80.2	79.8	82.0
3 (P + I +M)	16	70.6	76.8	85.0	87.2	90.0
3 (P + I +M)	32	75.6	78.6	83.2	88.0	89.6
3 (P + I +M)	64	74.6	78.0	83.6	88.2	90.6
5 (AF)	4	55.8	58.0	56.4	60.2	57.6
5 (AF)	8	70.6	73.8	77.8	83.8	85.4
5 (AF)	16	75.8	78.2	84.4	86.6	89.0
5 (AF)	32	77.2	80.8	82.4	87.8	91.8
5 (AF)	64	71.6	76.8	84.2	86.6	92.2
6 (P + AF)	4	51.4	53.6	61.4	54.4	53.8
6 (P + AF)	8	66.2	74.4	79.0	83.8	87.0
6 (P + AF)	16	72.2	79.8	84.0	85.8	91.0
6 (P + AF)	32	72.8	78.6	84.6	87.0	89.6
6 (P + AF)	64	74.4	82.6	83.0	88.4	89.8

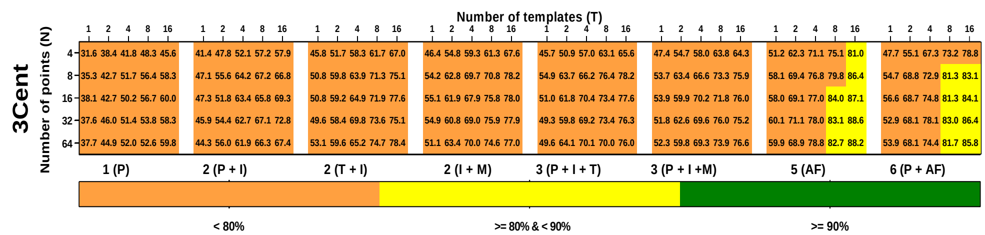
< 80% | >= 80% And < 90% | >= 90%

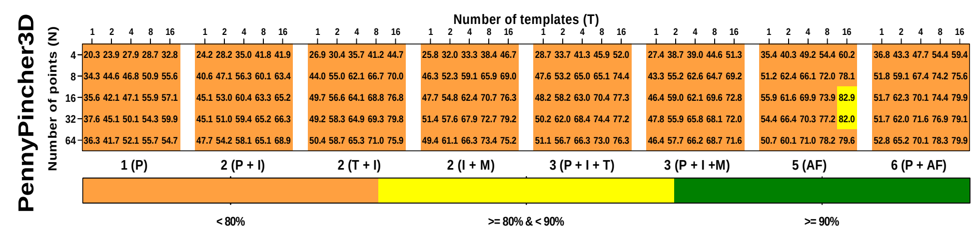
C.3 Recognition Rates Tables for the Jackknife-LM

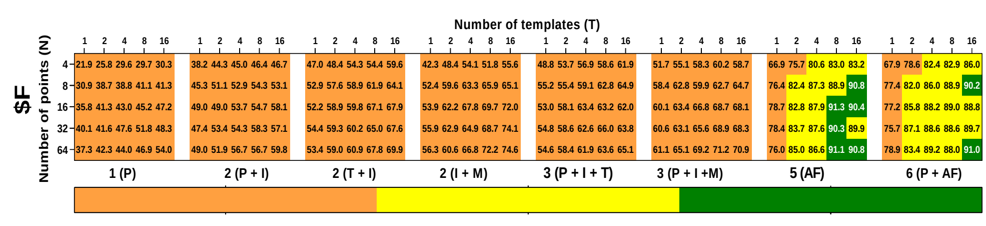












C.4 The ranking of the recognizers based on the best individual rates by articulation for SHREC2019

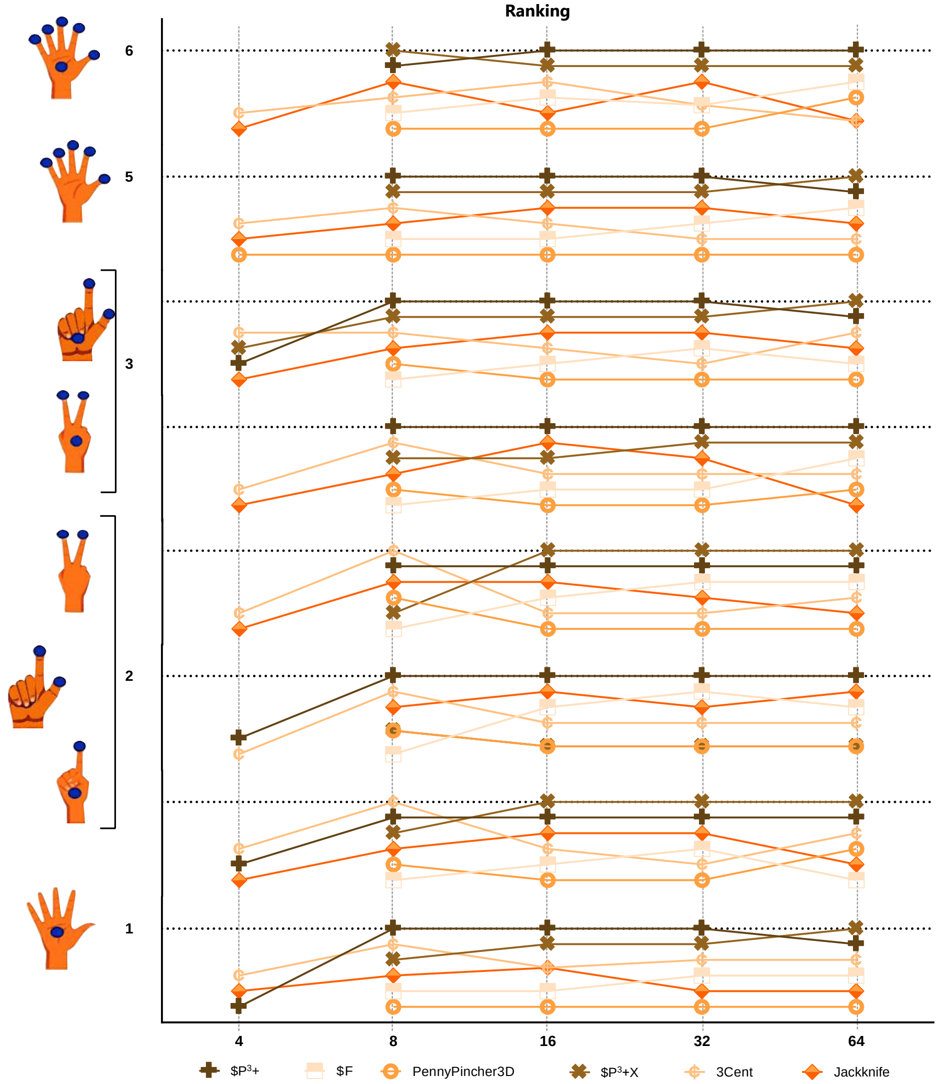


Fig. 11: Ranking of the best individual recognition rates above 80% by the number of joints (A) and the number of points (N) for the SHREC2019 dataset.

References

1. Abraham, L., Urru, A., Norman, N., Wilk, M.P., Walsh, M.J., O'Flynn, B.: Hand Tracking and Gesture Recognition Using Lensless Smart Sensors. Sensors **18**(9), 2834 (2018). https://doi.org/10.3390/s18092834
2. Akl, A., Valaee, S.: Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation, & compressive sensing. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 2270–2273 (April 2010). https://doi.org/10.1109/ICASSP.2010.5495895
3. Aliofkhazraei, M., Ali, N.: Recent Developments in Miniaturization of Sensor Technologies and Their Applications. In: Comprehensive Materials Processing, pp. 245–306. Elsevier, Oxford (2014). https://doi.org/10.1016/B978-0-08-096532-1.01309-1
4. Anthony, L., Wobbrock, J.O.: A Lightweight Multistroke Recognizer for User Interface Prototypes. In: Proceedings of Graphics Interface 2010. pp. 245–252. GI '10, Canadian Information Processing Society, Toronto, Ont., Canada, Canada (2010), https://dl.acm.org/doi/10.5555/1839214.1839258
5. Anthony, L., Wobbrock, J.O.: $N-ProTractor: A Fast and Accurate Multistroke Recognizer. In: Proceedings of Graphics Interface 2012. pp. 117–120. GI '12, Canadian Information Processing Society, Toronto, Ont., Canada, Canada (2012), https://dl.acm.org/doi/10.5555/2305276.2305296
6. Aquino, N., Vanderdonckt, J., Pastor, O.: Transformation templates: adding flexibility to model-driven engineering of user interfaces. In: Shin, S.Y., Ossowski, S., Schumacher, M., Palakal, M.J., Hung, C. (eds.) Proceedings of the 2010 ACM Symposium on Applied Computing (SAC), Sierre, Switzerland, March 22-26, 2010. pp. 1195–1202. ACM (2010). https://doi.org/10.1145/1774088.1774340
7. Benitez-Garcia, G., Haris, M., Tsuda, Y., Ukita, N.: Finger Gesture Spotting from Long Sequences Based on Multi-Stream Recurrent Neural Networks. Sensors **20**(2), 528 (2020). https://doi.org/10.3390/s20020528
8. Brunelli, R.: Template Matching Techniques in Computer Vision: Theory and Practice. John Wiley & Sons, New York (March 2009)
9. Caputo, F.M., Burato, S., Pavan, G., Voillemin, T., Wannous, H., Vandeborre, J.P., Maghoumi, M., Taranta II, E.M., Razmjoo, A., LaViola Jr., J.J., Manganaro, F., Pini, S., Borghi, G., Vezzani, R., Cucchiara, R., Nguyen, H., Tran, M.T., Giachetti, A.: Online Gesture Recognition. In: Biasotti, S., Lavoué, G., Veltkamp, R. (eds.) Eurographics Workshop on 3D Object Retrieval. pp. 93–102. The Eurographics Association (2019). https://doi.org/10.2312/3dor.20191067
10. Caputo, F.M., Prebianca, P., Carcangiu, A., Spano, L.D., Giachetti, A.: A 3 Cent Recognizer: Simple and Effective Retrieval and Classification of Mid-Air Gestures from Single 3D Traces. In: Proceedings of the Conference on Smart Tools and Applications in Computer Graphics. p. 9–15. STAG '17, Eurographics Association, Goslar, DEU (2017). https://doi.org/10.2312/stag.20171221
11. Coyette, A., Schimke, S., Vanderdonckt, J., Vielhauer, C.: Trainable sketch recognizer for graphical user interface design. In: Baranauskas, M.C.C., Palanque, P.A., Abascal, J., Barbosa, S.D.J. (eds.) Human-Computer Interaction - INTERACT 2007, 11th IFIP TC 13 International Conference, Rio de Janeiro, Brazil, September 10-14, 2007, Proceedings, Part I. Lecture Notes in Computer Science, vol. 4662, pp. 124–135. Springer (2007). https://doi.org/10.1007/978-3-540-74796-3_14
12. Davis, A.: Getting Started with the Leap Motion SDK (Feb 2014), https://blog.leapmotion.com/getting-started-leap-motion-sdk/

13. Ferrer, G., Sanfeliu, A.: Comparative Analysis of Human Motion Trajectory Prediction Using Minimum Variance Curvature. In: Proceedings of the 6th International Conference on Human-Robot Interaction. p. 135–136. HRI '11, Association for Computing Machinery, New York, NY, USA (2011). https://doi.org/10.1145/1957656.1957698
14. Filho, I.A.S., Chen, E.N., da Silva Junior, J.M., da Silva Barboza, R.: Gesture Recognition Using Leap Motion: A Comparison between Machine Learning Algorithms. In: ACM SIGGRAPH 2018 Posters. SIGGRAPH '18, Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3230744.3230750
15. Khalaf, A.S., Alharthi, S.A., Dolgov, I., Toups, Z.O.: A Comparative Study of Hand Gesture Recognition Devices in the Context of Game Design. In: Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces. p. 397–402. ISS '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3343055.3360758
16. Khan, R.Z.: Comparative Study of Hand Gesture Recognition System. In: Computer Science & Information Technology. vol. 2, pp. 203–213 (07 2012). https://doi.org/10.5121/csit.2012.2320
17. Leach, R.J.: Introduction to Software Engineering, Second Edition. Chapman & Hall/CRC, 2nd edn. (2016)
18. Magrofuoco, N., Pérez-Medina, J.L., Roselli, P., Vanderdonckt, J., Villarreal, S.: Eliciting Contact-Based and Contactless Gestures with Radar-Based Sensors. IEEE Access **7**, 176982–176997 (2019). https://doi.org/10.1109/ACCESS.2019.2951349
19. Magrofuoco, N., Roselli, P., Vanderdonckt, J.: Two-dimensional stroke gesture recognition: A survey. ACM Comput. Surv. **54**(7), 155:1–155:36 (2022). https://doi.org/10.1145/3465400, https://doi.org/10.1145/3465400
20. Marin, G., Dominio, F., Zanuttigh, P.: Hand Gesture Recognition with Jointly Calibrated Leap Motion and Depth Sensor. Multimedia Tools Appl. **75**(22), 14991–15015 (November 2016). https://doi.org/10.1007/s11042-015-2451-6
21. Nielsen, J.: Usability Engineering. Interactive Technologies, Elsevier Science (1994)
22. Ousmer, M., Sluÿters, A., Magrofuoco, N., Roselli, P., Vanderdonckt, J.: Recognizing 3D Trajectories as 2D Multi-Stroke Gestures. Proceedings of the ACM on Human-Computer Interaction **4**(ISS) (2020). https://doi.org/10.1145/3427326
23. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. **43**(1), 1–54 (2015). https://doi.org/10.1007/s10462-012-9356-9
24. Taranta, II, E.M., LaViola, Jr., J.J.: Penny Pincher: A Blazing Fast, Highly Accurate $-family Recognizer. In: Proceedings of the 41st Graphics Interface Conference. pp. 195–202. GI '15, Canadian Information Processing Society, Toronto, Ont., Canada, Canada (2015), https://dl.acm.org/doi/abs/10.5555/2788890.2788925
25. Taranta II, E.M., Samiei, A., Maghoumi, M., Khaloo, P., Pittman, C.R., LaViola Jr., J.J.: Jackknife: A Reliable Recognizer with Few Samples and Many Modalities. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 5850–5861. CHI '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3025453.3026002
26. Vanderdonckt, J., Roselli, P., Pérez-Medina, J.L.: !FTL, an Articulation-Invariant Stroke Gesture Recognizer with Controllable Position, Scale, and Rotation Invariances. In: Proc. of ICMI '18. p. 125–134. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3242969.3243032

27. Vatavu, R.D.: Improving Gesture Recognition Accuracy on Touch Screens for Users with Low Vision. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 4667–4679. CHI '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3025453.3025941
28. Vatavu, R.D., Anthony, L., Wobbrock, J.O.: Gestures As Point Clouds: A $P Recognizer for User Interface Prototypes. In: Proceedings of the 14th ACM International Conference on Multimodal Interaction. pp. 273–280. ICMI '12, ACM, New York, NY, USA (2012). https://doi.org/10.1145/2388676.2388732
29. Vatavu, R.D., Anthony, L., Wobbrock, J.O.: $Q: A Super-quick, Articulation-invariant Stroke-gesture Recognizer for Low-resource Devices. In: Proc. of MobileHCI '18. pp. 23:1–23:12. ACM, New York, NY, USA (2018). https://doi.org/10.1145/3229434.3229465
30. Wang, A., Chen, G., Yang, J., Zhao, S., Chang, C.Y.: A Comparative Study on Human Activity Recognition Using Inertial Sensors in a Smartphone. IEEE Sensors Journal **16**, 1–1 (06 2016). https://doi.org/10.1109/JSEN.2016.2545708
31. Wang, C., Liu, Z., Chan, S.C.: Superpixel-Based Hand Gesture Recognition With Kinect Depth Camera. IEEE Transactions on Multimedia **17**(1), 29–39 (2015). https://doi.org/10.1109/TMM.2014.2374357
32. Wobbrock, J.O., Wilson, A.D., Li, Y.: Gestures Without Libraries, Toolkits or Training: A $1 Recognizer for User Interface Prototypes. In: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology. pp. 159–168. UIST '07, ACM, New York, NY, USA (2007). https://doi.org/10.1145/1294211.1294238
33. Yasen, M., Jusoh, S.: A systematic review on hand gesture recognition techniques, challenges and applications. PeerJ Computer Science **5**, e218 (September 2019). https://doi.org/10.7717/peerj-cs.218
34. Zengeler, N., Kopinski, T., Handmann, U.: Hand Gesture Recognition in Automotive Human-Machine Interaction Using Depth Cameras. Sensors **19**(1), 59 (2019). https://doi.org/10.3390/s19010059